Delphi

บุญเลิศ เอี่ยมทัศนา

โปรแกรมด้วย Delphi : คอมโปเนนต์ใน Delphi 2.0

นำคอมโปเนนต์ใหม่ในหน้า Win 95 มาแนะนำ และจะขอแสดงตัวอย่างการใช้เพียงบางตัวเท่านั้น

ยุคของวินโดวส์ 95 คือยุคของแอพพลิเคชัน 32 บิต ดังที่บอร์แลนด์ได้ออกเดลไฟ 2.0 มาแทนเดลไฟ 1.0 เพื่อใช้กับวินโดวส์ 95 และ NT

หากต้องการจัดหา

ผมได้เปรียบเทียบเดลไฟ 2.0 กับ 1.0 ไปแล้วในฉบับก่อน ตอนนี้ได้ของจริงมาแล้วจึงขอนำมากล่าวเสริมสำหรับท่านที่ต้องการจัดหา

ก็ต้องขอย้อนกล่าวว่า เมื่อบอร์แลนด์ออก Delphi รุ่นแรก ไม่มีชื่อรุ่น มาอ้างถึงเมื่อออกรุ่น 2.0 แล้วว่า 1.0 และออกมา 2 ชุด คือ Delphi (ไม่มีสร้อยต่อ) ต่อมาเรียกกันว่าชุด Desktop กับชุด Client/Server ซึ่งผมจะขอเรียกเป็น C/S ชุด C/S ก็คือชุด Desktop บวก มีข้อแตกต่างกันในเรื่องงานฐานข้อมูลเท่านั้น คือ

- ชุด Desktop 1.0 ให้สามารถพัฒนาโปรแกรมฐานข้อมูล กับฐานข้อมูลภายใน (local database) และนำไปติดตั้งกับเครื่องของผู้ใช้ได้ และให้สามารถพัฒนาโปรแกรมกับฐานข้อมูลภายนอก (remote database) ได้ด้วย แต่นำไปติดตั้งให้แก่ผู้ใช้ไม่ได้

- ชุด C/S 1.0 ที่เพิ่มมาคือให้สามารถนำไปติดตั้งให้แก่เครื่องของผู้ใช้ เพื่อการใช้ฐานข้อมูลภายนอก เช่นฐานข้อมูลที่อยู่บนเครื่องที่ใช้ระบบจัดการ UNIX และระบบจัดการฐานข้อมูล Oracle เป็นต้น ซึ่งจะขอเรียกเป็นเครื่องนอก

สำหรับเดลไฟ 2.0 ได้พัฒนาต่อจากเดลไฟ 1.0 มากมาย ที่เป็นหลักคือเป็นแอพพลิเคชัน 32 บิต และใช้สิ่งแวดล้อมของวินโดวส์ 95 ผมจะนำมากล่าวในแง่ของการจัดหาครับ คือ ถ้าจะใช้เพื่อการเขียนโปรแกรมทั่วไป ชุดใดก็ได้ ซึ่งก็คงต้องเป็นชุด Desktop เพราะถูกกว่าเพื่อน แต่ถ้าจะนำมาใช้ในงานฐานข้อมูล ซึ่งก็คงต้องคำนึงถึงราคาด้วย คือ

- ชุด Desktop 2.0 คือชุด Desktop 1.0 ลบ คือพัฒนาและติดตั้งได้เฉพาะกับงานฐานข้อมูลภายในเท่านั้น

- ชุด Developer 2.0 คือชุด Desktop 1.0 บวก ที่บวกมาคือสิ่งที่ช่วยในการพัฒนาโปรแกรม

- ชุด C/S 2.0 คือชุด C/S 1.0 บวก เช่นกันครับ ที่บวกมาคือสิ่งที่จะช่วยในการพัฒนาระบบงานฐานข้อมูลภายนอก ที่ผมเห็นว่าสำคัญคือ InterBase NT ให้สามารถทดสอบการติดตั้งกับฐานข้อมูลภายนอกได้ โดยยังไม่ต้องติดต่อกับเครื่องนอกจริง ๆ

นอกจากนี้ในรุ่น 2.0 จะแถมรุ่น 1.0 ให้มาด้วย

ภายในกับภายนอก

คงจะพอใช้ในการจัดหาได้ แต่ที่ผมยังหาคำตอบที่ชัดเจนไม่ได้ คือ ในเรื่องของฐานข้อมูลภายนอก ในคู่มือของเดลไฟและในหนังสือต่าง ๆ (ที่กล่าวถึงการโปรแกรมด้วยเดลไฟ) จะกล่าวไว้อย่างชัดเจน ส่วนในเรื่องของฐานข้อมูลภายใน ในคู่มือของเดลไฟจะใช้คำว่า single machine ส่วนในหนังสือต่าง ๆ ไม่เอ่ยถึง

ที่ผมสงสัยคือ กับเครื่องที่ต่อกันด้วย LAN โดยเฉพาะอย่างยิ่งที่ใช้วินโดวส์ 3.11 หรือวินโดวส์ 95 จะใช้ฐานข้อมูลร่วมกันได้ไหม ที่ผมทดลองใช้คือเมื่อใช้ฐานข้อมูลตามแบบของ dBase ใช้ได้ แต่กับ Paradox ไม่ได้ แต่เป็นเครื่องที่ติดตั้ง Desktop 1.0 ทั้งคู่ (ทำงานเครื่องหนึ่ง ดูโปรแกรมเก่าหรือ Help อีกเครื่องหนึ่ง)

เพื่อให้ได้คำตอบที่ค่อนข้างแน่ชัด ผมจะต้องมีเครื่องอีก 2 เครื่อง ไม่ติดตั้งเดลไฟทั้งคู่ และกับวินโดวส์ 95 กับเดลไฟ 2.0 อีก จึงต้องเตรียมเครื่องก่อน แต่ผมไม่คิดจะมีเครื่องทั้งหมดนี้หรอกครับ ผมใช้วิธีเปลี่ยนฮาร์ดดิสก์

คอมโปเนนต์ใหม่

ในฉบับนี้จะขอนำคอมโปเนนต์ใหม่ในหน้า Win 95 มาแนะนำ และจะขอแสดงตัวอย่างการใช้เพียงบางตัวเท่านั้นคือ

1. TabControl เพื่อใช้แทน Tabset คือใช้เป็นที่คั่นหน้า

2. PageControl เพื่อใช้แทน TabbedNoteBox คือเป็นแฟ้มที่มีที่คั่นหน้าในตัว

3. TreeView, ListView และ ImageList เป็นคอมโปเนนต์ใหม่ให้ใช้งานร่วมกัน ดังเช่นที่ในวินโดวส์ 95 ใช้ทำเป็น Windows Explorer ดังในรูปที่ 1 ตามที่จะแสดงตัวอย่างต่อไป

4. HeaderControl ใช้แทน Header ของเดิม เพื่อใช้เป็นชื่อหัวข้อ เช่นของกรอบตาราง

5. RichEdit คือกรอบ Edit ใหม่ที่ใช้ทำ WordPad ในวินโดวส์ 95 ยังไม่ได้ลองโปรแกรมครับ จึงยังไม่ทราบว่าจะมีขีดความสามารถทางด้านภาษาไทยแตกต่างกับ WordPad Thai อย่างไร

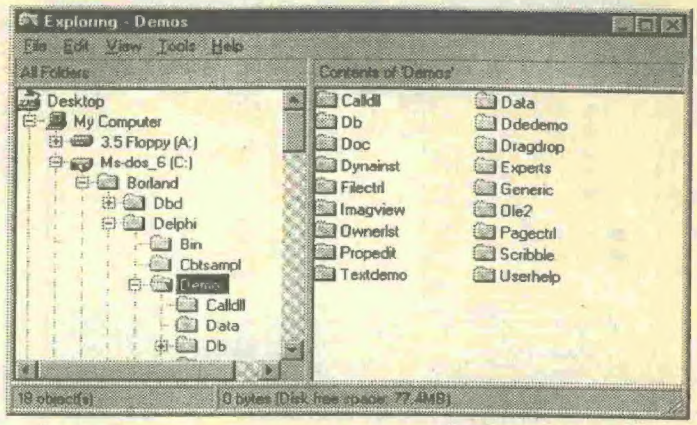
6. StatusBar ใช้เป็น Statusbur เพื่อแสดงสถานะการทำงานของโปรแกรม ปกติให้อยู่ที่ขอบด้านล่าง

7. TrackBar ใช้แทน ScrollBar

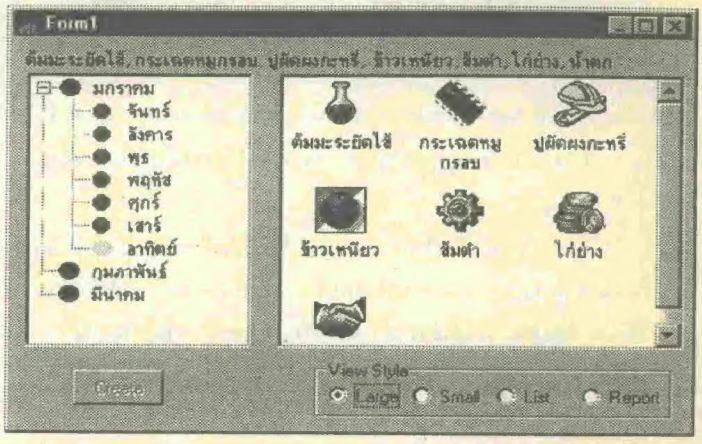
8. Progressbar ของเดิมคือ Gauage เป็นคอมโปเนนต์ตัวอย่างในหน้า Samples

9. UpDown ของเดิมมี SpinButton กับ SpinEdit สำหรับ Updown หากใช้ตัวเดียว คือ SpinButton หากใช้ร่วม (associate) กับกรอบ Edit คือ SpinEdit

10. Hotkey เพื่อให้ผู้ใช้กำหนดคีย์ด่วน (hotkey) และขอแก้ความเข้าใจซึ่งอาจผิดของผมในฉบับก่อนคือ ผมทดลองใช้ได้กับเมนูเท่านั้น



รูปที่ 1 หน้าต่างของ Windows Explorer ซึ่งแบ่งออกเป็น 2 บาน บานซ้ายคือ TreeView บานขวาคือ ListView ส่วนภาพที่แสดงเป็นไอคอนจะเก็บอยู่ใน ImageList



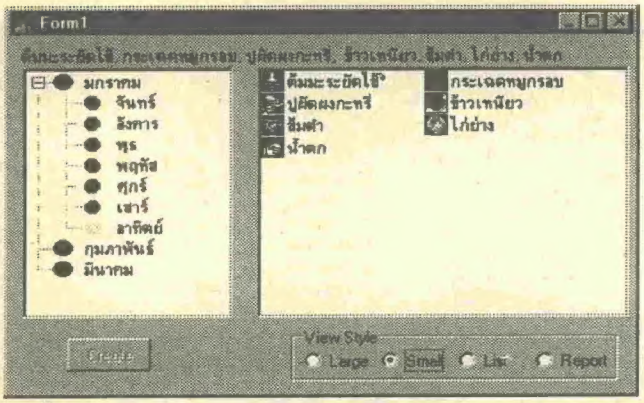
รูปที่ 2 การแสดงรายการอาหารของโปรแกรม ViewMenu ที่ให้แสดงไอคอนใหญ่

โปรแกรม ViewMenu

เพื่อแสดงการใช้ TreeView, ListView และ ImageList โดยจะให้แสดงรายการอาหาร (ในกรอบขวา) ของวัน และของเดือน (ในกรอบซ้าย) ตามแต่ผู้ใช้จะเลือก ดังเช่น

ในรูปที่ 2 เป็นรายการอาหารของเดือนมกราคม วันอาทิตย์ โดยให้แสดงรายการอาหารด้วยไอคอนขนาดใหญ่ตามที่เลือกปุ่ม Large (ภาพของไอคอนควรจะเข้ากับชื่ออาหาร แต่ฝีมือวาดภาพผมไม่มี จึงยืมเขามาใช้)

โดยที่ภาพรายการอาหารในกรอบทางขวาให้แสดงด้วย ListView ซึ่งอาจให้แสดงด้วยไอคอนขนาดเล็กได้ดังรูปที่ 3 โดยเลือกปุ่ม Small



รูปที่ 3 แสดงรายการอาหารด้วยไอคอนเล็ก

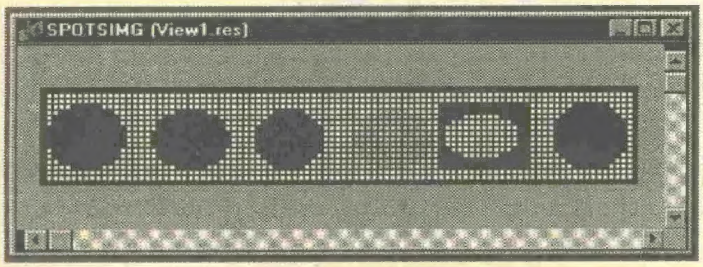
ลิสติงที่ 1

```
implementation

{$R *.DFM}
{$R view1.res}

procedure TForm1.FormCreate(Sender: TObject);
begin
  oMnuList := TStringList.Create;
  img1Tree.ResourceLoad(rtBitmap, 'SPOTSIMG', clOlive);
end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  oMnuList.Free;
end;
```



รูปที่ 4 ภาพไอคอนของไอคอน 6 ตัว ที่ให้รวมอยู่ในรูปเดียวกัน

ส่วนทางด้านซ้ายให้แสดงหัวข้อใหญ่คือชื่อเดือนและหัวข้อรองคือชื่อวัน ด้วย TreeView ส่วนที่เห็นเป็นจุดกลมความจริงแล้วคือไอคอน (ไม่มีฝีมือวาดเช่นกัน) ซึ่งควรเป็นภาพสัญลักษณ์ของเดือนและวัน และควรมี 2 ชุด คือเมื่อถูกเลือกกับไม่ถูกเลือก

ดังที่จะแสดงการใช้คอมโปเนนต์ทั้งสามในโปรแกรมนี้ ตามที่จะกล่าวในแต่ละหัวข้อต่อไป

เตรียมภาพไอคอน

แต่ต้องขอเตรียมภาพไอคอนก่อนครับ ตามที่กล่าวมาต้องเตรียมภาพไอคอน 3 ชุดคือ

- เพื่อใช้เป็นไอคอนของหัวข้อใหญ่และหัวข้อรอง
- เพื่อใช้เป็นไอคอนใหญ่ของรายการ (อาหาร)
- เพื่อใช้เป็นไอคอนเล็กของรายการ

เมื่อเตรียมแล้วจะต้องเก็บไว้ใน ImageList ซึ่งเก็บภาพได้ 2 แบบคือ

- เก็บเป็นแต่ละไอคอน ซึ่งจะต้องให้ค่าแก่ ImageList ในขณะเตรียมโปรแกรมเท่านั้น
- เก็บเป็นภาพรวม คือในการวาดภาพให้วาดภาพไอคอนทั้งหมดในรูปเดียวกันดังในรูปที่ 4 เรียงลำดับจากซ้ายไปขวา ลำดับที่คือ "ค่าอินเด็กซ์ของไอคอน" ตามที่จะใช้อ้างถึงต่อไป ส่วนการให้ค่าแก่ ImageList ให้ได้ทั้งในขณะเตรียมโปรแกรมและขณะรัน

จะแสดงตัวอย่างทั้งหมด โดยจะขอนำภาพ (ไอคอนและบิตแม็พ) ของเดลไฟมาใช้ โดยสมมติว่าได้วาดเก็บไว้แล้ว ส่วนภาพไอคอนของหัวข้อใหญ่และหัวข้อรองจะวาดรวมเป็นภาพเดียวกัน ดังในรูปที่ 4 และเพราะจะแสดงการให้ค่าในโปรแกรม จึงต้องทำเป็นรีซอร์สเก็บไว้ในไฟล์ .res

จะดำเนินการได้ดังนี้ โดยที่การดำเนินการในข้อหนึ่งจะเป็นผลให้ดำเนินการในข้อต่อไป คือ

1. เปิดใช้ Image Editor โดยเลือกรายการ Tools/Image Editor
2. เลือกรายการ File/New/Resource File
3. เลือกรายการ Resource/New/Bitmap
4. ให้ค่า Width เป็น 16x6 = 96 และค่า Height เป็น 16 จะได้ชื่อบิตแม็พรีซอร์สเป็น Bitmap1 ให้ป้อนแก้เป็น SPOTSIMG
5. ดับเบิลคลิ๊กที่ SPOTSIMG จะแสดงหน้าต่างสำหรับวาดภาพ แต่อาจไม่เห็นให้ขยายหน้าต่าง Image Editor
6. เลือกรายการ View/Zoom In เพื่อขยายพื้นที่วาดภาพ แล้ววาดรูปวงกลมสี น้ำเงิน แดง เขียว เหลือง ขาว และดำ รวม 6 วง ดังในรูปที่ 4 (จะวาดรูปใดก็ได้ครับให้มีความหมายเป็นปกติกับเลือกเป็นคู่ ๆ)
7. เลือกรายการ File/Save As เพื่อเก็บลงไฟล์โดยให้ชื่อไฟล์ว่า view1.res ในไดเรคทอรีเดียวกับที่จะเก็บโปรแกรม

ImageList

ในโปรแกรมนี้จะใช้ ImageList 3 ตัว จึงขอให้กำหนดคอมโปเนนต์เป็น ImageList รวม 3 ตัว ให้ชื่อ (ให้ค่าแก่ Name) เพื่อจะใช้เก็บภาพไอคอนดังนี้

1. **img1Tree** เพื่อเก็บภาพไอคอนของหัวข้อใหญ่และหัวข้อรองตามที่เตรียมไว้ในรีซอร์ส ซึ่งจะให้ค่าในโปรแกรมได้ ตามโปรแกรมในลิสติงที่ 1
2. **img1Small** เพื่อเก็บภาพไอคอนขนาดเล็กของรายการอาหาร ซึ่งจะให้ค่าในขณะเตรียมโปรแกรมได้ดังนี้

- คลิ๊กเม้าส์ขวาที่ imglSmall
- เลือกรายการ ImageList Editor
- เลือกปุ่ม Add จะแสดงไฟล์ไดอะล็อกบ็อกซ์ ให้เลือกไดเรคทอรี เป็น

```
C:\Program Files\Borland\Delphi 2.0\Image\Icons
```

- เลือกไฟล์ที่ลงท้ายด้วยเลข 16 ไฟล์แรกคือ Chem16
- ทำซ้ำ Add จนเลือกหมดทุกไฟล์แล้วเลือกปุ่ม OK

3. imglLarge เพื่อเก็บภาพไอคอนขนาดใหญ่ของรายการอาหาร ด้วยวิธีการเดียวกับของ imglSmall แต่เนื่องจากจะเป็นภาพขนาด 32x32 ก่อนโหลดภาพขอให้แก้ค่าพรอพเพอร์ตี้ส์ Height และ Width เป็น 32 ทั้งคู่ ส่วนภาพคือภาพที่เหลือในไดเรคทอรี Icons

เตรียมข้อมูล

สำหรับข้อมูลที่จะให้แสดงเป็น หัวข้อใหญ่ หัวข้อรอง และรายการ สุดแต่ผู้เขียนโปรแกรมจะออกแบบ สำหรับโปรแกรมนี้ได้ออกแบบไว้ดังในลิสติงที่ 2 กล่าวคือ ให้

1. ป้อนชื่อหัวข้อใหญ่ (ชื่อเดือน) ชิดซ้าย
2. ป้อนชื่อหัวข้อรอง (ชื่อวัน) โดยเว้นวรรค 1 วรรค
3. ป้อนชื่อรายการรวมทั้งเมื่อขึ้นบรรทัดใหม่ด้วย โดยกดคีย์ Tab 1 ครั้ง

เก็บไว้ในไฟล์ foodmenu.txt ในไดเรคทอรีเดียวกับโปรแกรม

ลิสติงที่ 2

```
มกราคม
 จันทร์  ต้มยำกุ้ง, คะน้าน้ำมันหอย, ปลาทอด
 อังคาร  แกงเขียวหวานไก่, กุ้งชุบแป้งทอด, ผักบุ้งไฟแดง
 พุธ     ไข่ยัดไส้, แกงจืดเต้าหู้, ผัดพริกหยวก
 พฤหัส  ผัดเนื้อน้ำมันหอย, แกงเลียง, ย่าปลากระป๋อง
 ศุกร์   ปลาช่องแปะซะ, แกงป่า, ผัดถั่วงอก
 เสาร์   แกงแดงเนื้อ, ไข่ลูกเขย, ผัดผักรวมมิตร
 อาทิตย์ ต้มมะระยัดไส้, กระเฉดหมูกรอบ, ปูผัดผงกะหรี่,
         ข้าวเหนียว, ส้มตำ, ไก่ย่าง, น้ำตก
กุมภาพันธ์
มีนาคม
```

ลิสติงที่ 3

```
const  chTab = #9;  chSpce = #32;  chCmma = ',';

function  GetToken(var txt : String) : String;
var  ic : Integer;
begin
  ic := Pos(chTab, txt);
  if  ic = 0  then  ic := Pos(chCmma, txt);
  if  ic > 0  then  Result := Copy(txt, 1, ic-1)
  else
    begin
    if  Length(txt) > 0  then
      begin  Result := txt;  txt := ''  end
    else  Result := ''
    end;
  if  txt <> ''  then
    txt := Copy(txt, ic+1, Length(txt));
end;

procedure TForm1.Button1Click(Sender: TObject);
var  oList : TStringList;  itm,ind : Integer;
     txt,adTxt : String;  oNde,oSub : TTreeNode;
begin
  TreeView1.ItemS.Clear;  oList := TStringList.Create;
  oList.LoadFromFile('foodmenu.txt');
   itm := 0;  oMnuList.Clear;
  while  itm < oList.Count  do
    begin
    txt := oList[itm];  inc(itm);
    if  txt[1] > chSpce  then
      begin
      oNde := TreeView1.ItemS.Add(TreeView1.Selected, txt);
      oNde.ImageInDex := 0;  oNde.SelectedIndex := 0;
      end
    else if  txt[1] = chSpce  then
      begin
      oSub := TreeView1.ItemS.AddChild(oNde, GetToken(txt));
      oMnuList.Add(txt);
      oSub.Data := Pointer(oMnuList.Count);
      oSub.ImageInDex := 2;  oSub.SelectedIndex := 3;
      end
    else
      begin
      txt[1] := chSpce;      ind := oMnuList.Count-1;
      adTxt := oMnuList[ind] + txt;
      oMnuList.Delete(ind);  oMnuList.Add(adTxt);
      end;
    end;
  oList.Free;   Button1.Enabled := False;
end;
```

TreeView

ขอให้กำหนดคอมโปเนนต์เป็น TreeView1 เป็นกรอบทางด้านซ้าย ในการใช้งานโดยทั่ว ๆ ไป ดังเช่นตามตัวอย่างนี้ มีค่าพรอพเพอร์ตี้ส์และค่าในพรอพเพอร์ตี้ส์ที่จะต้องให้ 3 ค่า ซึ่งให้ได้ทั้งในขณะเตรียมโปรแกรมและในขณะรัน จะเลือกให้ตามค่าที่เตรียมไว้คือ

1. Images จะต้องต่อเชื่อมเข้ากับ ImageList ที่เตรียมไว้ ของ TreeView1 คือ ImglTree โดยเลือกหัวข้อ Images ในหน้า Properties แล้วเลือกจากกรอบคอมโบ

2. Items คือชื่อหัวข้อหลักและรอง ซึ่งถ้าต้องการให้ค่าในขณะเตรียมโปรแกรมสามารถให้ค่าได้โดยตรง ส่วนการให้ค่าในโปรแกรมคือตามลิสติงที่ 3 คือเมื่อผู้ใช้เลือกปุ่ม Create จะมาทำงานที่โปรแกรมของกิจกรรม Button1Click โดยที่

- หัวข้อแต่ละหัวข้อคือโหนดหนึ่ง ๆ ซึ่งอาจมีโหนดลูกได้ เช่น มกราคม มีโหนดลูกเป็นวันจันทร์ ถึง อาทิตย์

- กิจกรรม Add จะผนวกโหนดในระดับเดียวกันต่อท้าย เช่น กุมภาพันธ์ ต่อท้ายมกราคม หรือ ศุกร์ ต่อ พฤหัส เป็นต้น

- กิจกรรม AddChild จะผนวกโหนดลูก ต่อท้ายโหนดลูกที่มีอยู่แล้ว (หากมี)

ในโปรแกรมจึงให้ดำเนินการดังนี้ ซึ่งมีเรื่องอื่นอยู่ด้วยตามที่จะกล่าวต่อไป คือ

ให้อ่านข้อมูลในไฟล์ foodmenu.txt มาเก็บไว้ใน oList เพื่อที่จะอ่านมาให้แก่ txt ครั้งละ 1 บรรทัด แล้วนำข้อความใน txt มาตรวจ กล่าวคือ

2.1 ถ้าเป็นชื่อเดือน (ป้อนชิดซ้ายทำให้อักขระแรกมีค่ามากกว่าวรรค) จะให้

- ผนวกเป็นหัวข้อใหญ่ ด้วย Add

- เก็บออบเจ็กต์ของโหนดนี้ไว้ใน oNde ซึ่งโหนดแรกคือเดือนมกราคม

2.2 ถ้าเป็นชื่อวัน (อักขระแรกเป็นวรรค) ให้

- ผนวกเป็นหัวข้อรองของโหนดใน oNde ด้วย AddChild

- โดยให้ GetToken ตัดคำที่เป็นชื่อวันให้

- เก็บข้อความที่เหลือซึ่งเป็นรายชื่ออาหาร ไว้เป็นรายการใน oMenuList และเก็บค่าอินเด็กซ์บวกหนึ่งคือค่า Count ของรายการ (อาหาร) ใน oMenuList ไว้ใน Data ของโหนดนั้นคือของวันนั้น จึงเท่ากับเก็บรายการอาหารของวันใดไว้ในวันนั้น

การให้ค่าแก่ Items มีเพียงเท่านี้ครับ โปรแกรมท่อนต่อไปคือกรณีที่ป้อนรายการอาหารมากกว่า 1 บรรทัด จึงนำมาต่อกัน

3. ค่าอินเด็กซ์ของไอคอน

มีค่าอินเด็กซ์ของไอคอนที่จะต้องให้ 3 ค่า คือ

- ImageIndex เพื่อแสดงสถานะปกติ

- SelectedIndex เพื่อแสดงสถานะเมื่อถูกเลือก

- StateIndex เป็นค่าอินเด็กซ์ของไอคอนใน ImageList ซึ่งจะต้องต่อเชื่อมคือให้ค่าแก่ StateImages ของ TreeView1 แยกต่างหาก ซึ่งในโปรแกรมนี้ไม่ได้นำมาใช้

คงให้ค่าอินเด็กซ์ของไอคอนแก่ ImageIndex กับ SelectedIndex เท่านั้น ในโปรแกรมของกิจกรรม Button1Click โดยที่

1. ของหัวข้อใหญ่ให้เป็นศูนย์ (วงกลมสีน้ำเงิน) ทั้งคู่ เพื่อบอกแก่ผู้ใช้ว่าเป็นหัวข้อที่ไม่มีรายการ ต่างกับของ Windows Explorer ที่ถือว่าโฟลเดอร์เป็นทั้งหัวข้อรองและรายการของโฟลเดอร์ในระดับบนเช่นเดียวกับไฟล์

2. ของหัวข้อรอง ให้เป็น 2 (วงกลมสีเขียว) และ 3 (วงกลมสีเหลือง) ตามลำดับ

โดยที่การกำหนดไอคอนอาจกำหนดให้เป็นกลุ่ม เช่น ของเดือนเหมือนกัน ของวันเหมือนกัน เช่นตามตัวอย่างนี้ หรือของเดือนแต่ละเดือน วันแต่ละวัน ไม่เหมือนกัน

TListView

กรอบ TListView จะแสดงรายการพร้อมด้วยไอคอน มีรูปแบบของการแสดงภาพ 4 รูปแบบด้วยการให้ค่าแก่ ViewStyle คือ

- vsIcon แสดงภาพไอคอนขนาดใหญ่ดังในรูปที่ 2

- vsSmallIcon แสดงภาพไอคอนขนาดเล็ก เรียงลำดับจากซ้ายไปขวาและจากบนลงล่าง ดังในรูปที่ 3

- vsList แสดงภาพไอคอนขนาดเล็ก เรียงลำดับจากบนลงล่างและจากซ้ายไปขวา

- vsReport แสดงภาพไอคอนขนาดเล็ก พร้อมด้วยรายละเอียดของรายการในรูปของตารางดังในรูปที่ 5

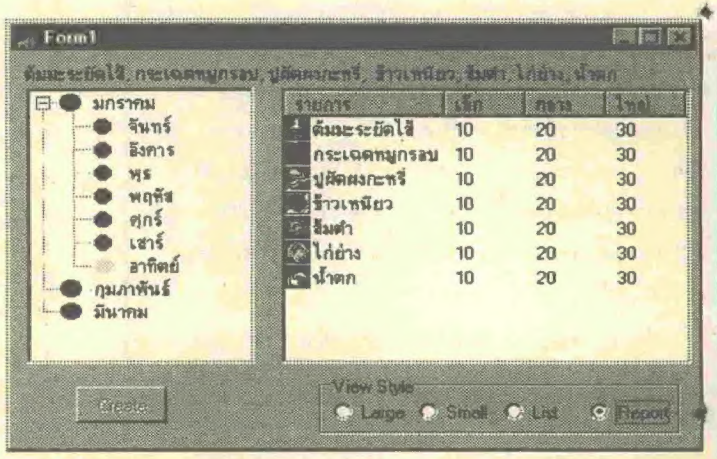
ขอให้กำหนดคอมโปเนนต์เป็น ListView1 คือกรอบทางขวา มีค่าพรอพเพอร์ตี้ส์ที่จะต้องให้ทั้งในขณะเตรียมโปรแกรมและในขณะรันดังนี้

1. Colums เป็นรายการชื่อคอลัมน์ ดังที่เห็นในรูปที่ 5 ซึ่งจะให้ค่าได้ดังนี้

- เลือกหัวข้อนี้แล้วเลือกปุ่มอิลิปต์ (...) จะแสดงไดอะล็อกบ็อกซ์ให้ป้อนชื่อคอลัมน์ โดย

- เลือกปุ่ม New แล้วป้อนข้อความว่า "รายการ" แล้วให้ค่า Width เป็น 100

- ให้ป้อนชื่อคอลัมน์ต่อ ๆ ไปคือ เล็ก กลาง และใหญ่ ด้วยวิธีเดียวกัน โดยให้ค่า Width เป็น 50 เท่ากันทั้งหมด



รูปที่ 5 รายการอาหารพร้อมด้วยราคา ที่ให้แสดงในรูปของตาราง

2. Images มีค่า Images ที่จะต้องให้ (ต่อเชื่อม) 2 ค่า คือ

- LargeImages ให้เป็น imglLarge
- SmallImages ให้เป็น imglSmall

3. Items คือค่ารายการ (อาหาร) สามารถให้ได้ทั้งในขณะเตรียมโปรแกรม หรือในขณะรันเช่นตามตัวอย่างนี้ คือในโปรแกรมของกิจกรรม TreeView1MouseDown ในลิสติงที่ 4 ซึ่งเมื่อผู้ใช้คลิ๊กเม้าส์ขาลงในกรอบ TreeView1 จะทำให้เกิดอีเวนต์นี้และมาทำงานที่กิจกรรมนี้ คือ

3.1 ตรวจว่าผู้ใช้คลิ๊กเม้าส์ตรงชื่อหัวข้อ ซึ่งอาจเป็นชื่อเดือนหรือชื่อวัน คือ htOnItem หรือไม่ ซึ่งถ้าไม่ตรงชื่อเช่นคลิ๊กตรงเครื่องหมายบวก จะให้กลับไป (ไม่ดำเนินการต่อ) เพื่อไม่ให้สับสน

3.2 ถ้าตรงชื่อหัวข้อ จะให้ออบเจ็กต์ของโหนดนั้นแก่ oNde แล้วตรวจค่าใน Data ถ้าเป็น nil คือศูนย์ แสดงว่าเป็นชื่อเดือน ให้กลับไปเช่นกัน กรณีที่เหลือคือเมื่อผู้ใช้คลิ๊กเม้าส์ตรงชื่อวัน

3.3 ให้อ่านรายการอาหารตามที่เก็บไว้ในวันตามที่กล่าวมาออกแสดง ดังที่เห็นอยู่เหนือกรอบทั้งสอง โดยให้เป็นค่า Caption ของ Label1 เพื่อการตรวจสอบเท่านั้น

3.4 ให้อ่านค่ารายการอาหารมาให้แก่ txt เพื่อจะแตกเป็นชื่ออาหารให้แก่ ListView1 โดย

- ยกเลิกรายการเดิมใน ListView1 (หากมี)
- ให้ GetToken ตัดชื่ออาหารให้ครั้งละ 1 ชื่อ ให้แก่ adTxt
- ผนวกไอเท็ม (รายการ) เปล่า ให้แก่ ListView1 ด้วย Add แล้วให้ค่าออบเจ็กต์ของไอเท็มแก่ oItm
- ให้ชื่ออาหารแก่ไอเท็ม และให้ค่าอินเด็กซ์ของไอคอน แก่ ImageIndex ของไอเท็มนั้น เริ่มจากศูนย์เป็นต้นไป จึงทำให้ได้ภาพไอคอนแตกต่างกัน

3.5 ใน ListView ถือว่ารายละเอียดของรายการเป็นรายการของ (SubItemS) ของรายการไอเท็ม ซึ่งในโปรแกรมนี้กำหนดให้มีรายการรอง 3 รายการ โดยสมมติว่าเป็นราคาของอาหารจาน เล็ก กลาง และใหญ่ เพื่อความง่ายจึงให้ค่าราคาไว้เป็น 10, 20 และ 30 เท่ากันหมด โดยใช้ Add ผนวกให้เป็นรายการรอง จึงได้ผลดังในรูปที่ 5 (มีรายละเอียดตามที่จะกล่าวต่อไป)

4. ViewStyle เพื่อการแสดงผลทั้งสี่แบบตามที่กล่าวมา ซึ่งอาจทำเป็นเมนูให้เลือกดังเช่นใน Windows Explorer หรือตามโปรแกรมนี้ให้ผู้ใช้เลือกจากกรอบรายการปุ่มวิทยุ โดย

4.1 กำหนดคอมโปเนนต์เป็น RadioGroup1 ดังที่เห็นอยู่ตรงมุมล่างขวา แล้วให้ค่า

- รายการปุ่มวิทยุ โดยให้แก่ ItemS เป็น Large, Small, List และ Report (หนึ่งคำต่อหนึ่งบรรทัด)
- ให้ค่า ItemIndex เป็น 0

4.2 ให้ค่า ViewStyle ของ ListView1 เป็น vsIcon เพื่อให้ตรงกัน

4.3 เมื่อผู้ใช้เลือกปุ่มวิทยุ จะมาทำงานที่กิจกรรม Radio Group1Click ในลิสติงที่ 4 ซึ่งได้ให้ค่าอินเด็กซ์ของปุ่มวิทยุที่ผู้ใช้เลือกแก่ ViewStyle ตามที่ได้จัดค่าไว้ให้ตรงกัน จะทำให้ ListView1 แสดงภาพตามรูปแบบที่ผู้ใช้เลือก

```
procedure TForm1.TreeView1MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
var  HT : THitTests; oNde : TTreeNode;  txt,adTxt : String;
     oItm : TListItem;   cnt : Integer;
begin
  HT := TreeView1.GetHitTestInfoAt(X,Y);
  if  not(htOnItem in HT) then  Exit;
  oNde := TreeView1.GetNodeAt(X,Y);
  if  oNde.Data = nil  then  Exit;
  Label1.Caption := oMnuList[Integer(oNde.Data) - 1];

  txt := oMnuList[Integer(oNde.Data) - 1];
  ListView1.ItemS.Clear;   cnt := 0;
  while  (txt <> '')  do
    begin
    adTxt := GetToken(txt);
    oItm := ListView1.ItemS.Add;
    oItm.CapTion := TrimLeft(adTxt);
    oItm.ImageIndex := cnt;  inc(cnt);
    oItm.SubItemS.Add('10');
    oItm.SubItemS.Add('20');
    oItm.SubItemS.Add('30');
    end;
end;


procedure TForm1.RadioGroup1Click(Sender: TObject);
begin
  ListView1.ViewStyle := TViewStyle(RadioGroup1.ItemIndex);
end;

end.
```

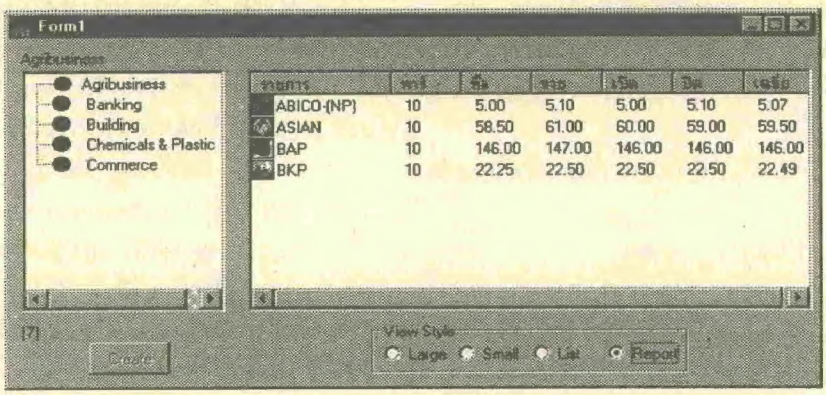
ลิสติงที่ 4

ลิสติงที่ 5

```
Root=Agribusiness, Banking, Building, Chemicals & Plastic, Commerce

Agribusiness=ABICO-(NP), ASIAN, BAP, BKP
Banking=BAY, BBC, BBL, BMB
Building=APC, ASTL-(XD), BPT-(NP), BSI
Chemicals & Plastic=TPI
Commerce=PE

ABICO-(NP)=1, 10, 5.00, 5.10, 5.00, 5.10, 5.07
ASIAN=5, 10, 58.50, 61.00, 60.00, 59.00, 59.50
BAP=3, 10, 146.00, 147.00, 146.00, 146.00, 146.00
BKP=7, 10, 22.25, 22.50, 22.50, 22.50, 22.49
```



รูปที่ 6 ผลการรันของโปรแกรมที่ให้แสดงข้อมูลตามลิสติงที่ 5

การจัดเก็บข้อมูล

การจัดเก็บข้อมูลขึ้นอยู่กับผู้เขียนโปรแกรมจะกำหนด ตามที่กล่าวมาเป็นวิธีหนึ่ง อีกวิธีหนึ่งซึ่งอาจจะป้อนข้อมูลยาก แต่จะใช้งานได้กว้างกว่าและอ่อนตัวกว่า คือตามลิสติงที่ 5 เพื่อการแสดงผลตามรูปที่ 6 กล่าวคือ

- Root เป็นคำที่ตั้งขึ้น เพื่อให้รายชื่อหัวข้อใหญ่ คือที่ใช้เครื่องหมาย = ตามด้วยรายชื่อหัวข้อใหญ่

- นำชื่อหัวข้อใหญ่ มาให้รายชื่อรายการ (คือชื่อหลักทรัพย์)

- นำชื่อรายการ มาให้ค่ารายละเอียด (คือราคาต่าง ๆ) แต่ตัวเลขจำนวนแรกคือค่าอินเด็กซ์ของไอคอน

ตามตัวอย่างนี้ไม่มีหัวข้อรอง ซึ่งถ้ามีและจะลึกกี่ชั้นก็ได้ คงใช้วิธีเดียวกัน และอาจเก็บค่าอินเด็กซ์ของไอคอนของหัวข้อต่าง ๆ ได้เช่นกัน ซึ่งต้องใช้เลข 2 จำนวน

ขอให้ทดลองเขียนโปรแกรมครับ จะเขียนได้สั้นกว่าของโปรแกรม ViewMenu

Help

ต้องขอบ่นครับ Help ของเดลไฟ 1.0 ว่าแย่แล้ว ของ 2.0 ยิ่งกว่า หาคำไม่ค่อยพบ ต้องดูในคู่มือคือ Reference Library Guide และดูเหมือนจะเป็นแฟชั่นของวินโดวส์ 95 บอกให้ติดต่อกับผู้ขาย พวกเดียวกับสมุดโทรศัพท์ บอกให้ไปรับเอง ประหยัดดี